



Czerepinski, P.J., & Bull, D.R. (1996). An Entropy Coder For Segmentation Based Video Compression. In *First International Workshop on Wireless Image/Video Communications - September 1996* (pp. 43 - 47). Institute of Electrical and Electronics Engineers (IEEE). <https://doi.org/10.1109/WIVC.1996.624641>

Publisher's PDF, also known as Version of record

Link to published version (if available):  
[10.1109/WIVC.1996.624641](https://doi.org/10.1109/WIVC.1996.624641)

[Link to publication record in Explore Bristol Research](#)  
PDF-document

Copyright © 1996 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

## University of Bristol - Explore Bristol Research

### General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:  
<http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/>

## AN ENTROPY CODER FOR SEGMENTATION BASED VIDEO COMPRESSION

*P.J. Czerepinski and D.R. Bull*

Image Communications Group, Centre for Communications Research,  
University of Bristol, Merchant Venturers Bldg., Woodland Rd., Bristol BS8 1UB, UK  
email: dave.bull@bristol.ac.uk p.j.czerepinski@bristol.ac.uk  
Tel. +44 117 9545195; Fax: +44 117 9255265

**Abstract** *Morphological segmentation has recognised advantages for video compression, especially at lower bit rates. A two stage approach has traditionally been employed to encode the interframe data produced by this algorithm: contour coding of the regions selected for transmission, followed by coding of the data within the regions. In this paper, an effective single stage conditional arithmetic coder is demonstrated to successfully accomplish this task.*

### 1 INTRODUCTION

Morphological segmentation has recently been proposed as an alternative method for coding interframe data in video sequences [1] and is being actively investigated in MPEG4. Poor correlation properties of the displaced frame difference data (DFD) often lead to ringing artefacts in the reconstruction if standard motion-compensated discrete cosine transform (DCT) coders are used. Morphological segmentation is a non-linear algorithm, which enables the selection or removal of frame-to-frame differences based on a shape/size criterion. Being a spatial approach, it offers a lack of ringing and edge preservation properties.



Figure 1. Original frame, DFD (absolute value) and update signals created by the morphological segmentation algorithm (black denotes inactive and white active areas). Top: 'Akiyo', bottom: 'Foreman'.

Morphological segmentation is applied to the DFD data, yielding an ‘update’ signal, which typically contains a few motion model failure areas (active areas), surrounded by zero valued pixels. These active areas are quantised and selected for transmission (figure 1.). The signal is usually coded losslessly by means of a sophisticated two stage algorithm: boundaries of the active areas are coded using chain codes [1][2], and then pixel values within the contours are encoded. The complexity of this approach results from its two-stage nature, and is augmented by the necessity of dealing with nested model failure areas. In this paper a simple yet effective algorithm is proposed for the entropy coding of the morphologically segmented DFD data.

## 2 ENTROPY CODER

The two-stage approach, discussed above, is replaced with a single stage coder. Although this implies that the non-active areas (typically more than 90% of the DFD pixels) must be coded along with the active ones, it will be shown that they can be coded using relatively few bits. This is accomplished by means of a two state conditional model, depending on the state  $S$  of the encoder:

$$S = \begin{cases} S1 & \text{if } \forall_v \in \{X, Y, Z\}, V = \text{OFF} \\ S2 & \text{if } \exists_v \in \{X, Y, Z\}, V = \text{ON} \end{cases} \quad (1)$$

where  $X, Y, Z$  are causal half-plane neighbours of the encoded pixel. An arithmetic coder has been employed [3], since it deals well with low entropy signals, facilitates adaptivity and offers a clear separation between the modelling and coding. The ‘Silent Voice’, ‘Claire’ and ‘Trevor’ sequences have been used for model initialisation rather than a uniform distribution. In addition, in the case of the adaptive coder, the rate at which symbols update the model has been empirically optimised. Experiments were performed with three types of adaptive approach:

- scheme A1: where the model is reset to the initial distribution at the beginning of every frame;
- scheme A2: where the model is built throughout the whole coding process, such that frame  $n$  is coded based on the statistics of  $n-1$  previously coded frames;
- scheme A3: in arithmetic coding, a representation of symbol probabilities is stored by coder’s model: e.g. in state  $S_x$  symbols  $\{k_1, k_2, \dots, k_r, \dots, k_N\}$  are assigned probabilities, corresponding to histogram values  $\{h_{x1}, h_{x2}, \dots, h_{xr}, \dots, h_{xN}\}$ . For efficient coding, the model should be able to trace the statistics of the source as closely as possible. This is accomplished by updating the model as symbols are coded, i.e. after coding the symbol  $k_r$ , in state  $S_x$ , the corresponding symbol histogram is set to  $\{h_{x1}, h_{x2}, \dots, h_{xr} + u, \dots, h_{xN}\}$ , where  $u$  is an arbitrary update term. Due to

implementation constraints, the sum  $\sum_{i=1}^N h_{xi}$  must not exceed some maximum value,  $H_{\max}$ . To

prevent this from happening, histogram entries  $h_{xi}$  are halved whenever their sum approaches  $H_{\max}$ . In schemes A1 and A2, coder models corresponding to states  $S1$  and  $S2$  were updated at an identical rate. Note, however, that due to the nature of the coded signal, the majority of pixels are encoded in state  $S1$  and only a few (typically 1-5%) in state  $S2$ . Intuitively, the statistics of the state that occurs less frequently should be updated faster. In order to achieve higher efficiency, the following approach was adopted: firstly, the value of the update rate  $u_1$ , corresponding to the state  $S1$ , was fixed and the update rate  $u_2$ , corresponding to the state  $S2$  was empirically optimised. The procedure was repeated, by varying  $u_1$  with  $u_2$  fixed. The optimisation is then complete, due to the independence of the states, i.e. the fact that if the coder is in state  $S1$  then it can not be in  $S2$  and vice versa. As with coder A1, the model is set to the initial distribution at the beginning of each frame.

### 3 CODING RESULTS

The coding algorithm employed consists of a wavelet intraframe mode (Le Gall's odd filters [4]) and a segmentation based interframe mode. In order to reduce the effect of artificial edges, occurring at boundaries of the transmitted update regions, a localised smoothing algorithm was applied as described in [5][6]. Table 1. shows arithmetic coder parameters. The results obtained using the 'Akiyo' (240×352×8bpp) and 'Foreman' (288×352×8bpp) sequences are summarised in tables 2 and 3.

Firstly, a fixed conditional coder was applied. The packing capabilities of this approach alone are superior than those of a simple 1<sup>st</sup> order coder. It is clear from tables 2 and 3 that schemes A1 and A2 outperform the fixed scheme. Also, A2 outperforms A1 in the case of the 'Akiyo' sequence and is outperformed by A1 in the case of 'Foreman'. The percentage difference, however, is minor, due to the relatively large frame size, which allows model adaptation within a single frame. In presence of channel errors, method A2 is likely to suffer from infinite error propagation. Thus, any advantage gained by using A2 will diminish as the bit error rate increases. Restarting the model at the beginning of every frame will limit error propagation, as long as the start of the frame can be correctly determined at the decoder. Therefore, scheme A1 is deemed a more attractive solution.

Of all the coders tested, A3 yields the best performance. Comparing A3 to A1, it is clear that bit rate savings have been achieved when coding the active regions. Indeed, the number of bits required to code the black areas is slightly higher in the case of A3 than in the case of A1. This merely means that not all the inactive pixels are encoded in state  $S_1$  and not all active pixels are coded in state  $S_2$ , and does not imply that the update rates used are not optimal.

coder	no. of states	symbols/state	$H_{\max}$	update rates	
				$S_1$	$S_2$
fixed	2	42	32767	0	0
A1	2	42	32767	110	110
A2	2	42	32767	50	50
A3	2	42	32767	25	900

Table 1. *Coder parameters.*

### 4 CONCLUSIONS

A candidate algorithm for entropy coding the update signal in segmentation-based video coders has been proposed. As can be verified, the cost of coding the inactive regions is between 20 and 30% of the total bit rate. The algorithm comprises a single entropy coding stage only and avoids the overhead of boundary coding. It thus offers an advantage of implementation simplicity and possible bit rate reduction. Work is ongoing to compare this approach with region coding (exact and approximate) in terms of performance and computational complexity.

One obvious disadvantage of the proposed method is the need to encode all-black frames, such as frames 1-4 of 'Akiyo'. This can be simply remedied by introducing a special 'empty frame' symbol.

Outstanding issues include: how fast should a model be updated and whether optimum update rates for an adaptive coder can be determined *a priori*. Intuitively, optimum update rates will depend

on the predictability of the signal coded and the proportion of symbols coded in different states. Note that the ratio  $u_1/u_2$  (coder A3) is roughly equal to that of the number of pixels coded in state S2 to the number of pixels coded in state S1.

frame	First order model	Conditional models							
		fixed		adaptive (A1)		adaptive (A2)		adaptive (A3)	
		i	a	i	a	i	a	i	a
1	0	370	0	211	0	211	0	212	0
2	0	370	0	211	0	210	0	212	0
3	0	370	0	211	0	210	0	212	0
4	0	370	0	211	0	210	0	212	0
5	2828	492	1000	380	910	374	920	388	849
6	3095	522	959	419	864	420	858	435	778
7	1708	454	480	318	458	323	480	327	456
8	1865	471	598	337	564	341	562	341	556
9	1137	417	368	273	352	272	340	275	313
10	1648	447	516	309	492	310	472	317	447
11	3167	579	1031	461	1048	476	1017	462	1023
12	2623	485	853	363	782	357	775	384	710
13	690	407	244	258	251	261	230	261	245
14	3122	550	950	432	947	438	931	439	934
15	3228	583	1092	463	1098	475	1073	460	1124
16	3862	611	1234	520	1225	511	1234	510	1222
17	2095	473	572	343	567	335	573	351	566
18	1026	433	321	286	322	288	312	285	328
19	7347	893	2619	845	2654	850	2612	845	2588
20	3838	626	1233	508	1247	500	1233	511	1233
21	6969	900	2634	835	2696	831	2659	835	2698
22	4320	692	1428	589	1493	586	1489	592	1430
23	3512	616	1134	510	1180	493	1183	502	1184
24	425	400	111	246	117	244	120	246	114
25	2466	548	823	419	812	407	810	416	815
26	2056	503	629	371	657	358	655	368	668
27	2698	569	883	435	924	427	917	433	930
28	7020	865	2592	794	2510	779	2501	803	2491
29	7571	886	2895	841	2812	844	2761	854	2766
30	4533	659	1568	567	1500	563	1473	574	1446
31	3161	561	1105	444	1089	440	1064	446	1108
$\Sigma$	88010	17122	29872	13410	29571	13344	29254	13508	29022
$\Sigma(b,w)$	88010	46994		42981		42598		42530	
[%]	187.3	100		91.46		90.65		90.50	

Table 2. Coding results for 'Akiyo' at 48 kbps, mean PSNR=35.85.  $i$  and  $a$  denote the number of bits required to encode the inactive and active areas respectively.

frame	First order model	Conditional models							
		fixed		adaptive (A1)		adaptive (A2)		adaptive (A3)	
		i	a	i	a	i	a	i	a
1	3019	583	761	433	646	425	694	455	517
2	10987	980	3449	1064	2999	1053	3102	1097	2693
3	13734	1099	4721	1256	4144	1274	4273	1251	3631
4	20423	1372	7908	1780	6246	1795	6290	1764	5701
5	14788	1246	4996	1444	4418	1457	4462	1412	4068
6	12357	1149	4248	1255	3716	1274	3733	1265	3450
7	12173	1064	4058	1161	3603	1160	3718	1184	3296
8	12055	1028	4038	1136	3126	1179	3171	1165	2789
9	17241	1215	7225	1481	5745	1490	5874	1469	5235
10	19727	1518	8125	1839	6858	1905	6970	1834	6583
11	19149	1580	7926	1877	6755	1915	6762	1825	6579
12	10735	1041	4051	1091	3582	1063	3648	1085	3447
13	21380	1614	9162	1997	7641	1969	7811	1991	7156
14	29711	2077	12288	2802	10341	2822	10450	2807	9872
15	17181	1416	7338	1659	6355	1671	6392	1653	6068
16	18846	1718	8001	1944	7294	1941	7328	1962	6893
17	16043	1464	6825	1624	5969	1627	5982	1630	5834
18	10127	954	4070	963	3619	972	3632	977	3328
19	9668	1006	3460	1008	3174	1049	3145	1016	2931
20	11392	1086	4837	1115	4378	1124	4362	1138	4237
21	5204	695	1903	616	1754	647	1701	627	1655
22	17661	1348	8103	1574	6511	1585	6551	1574	6232
23	12298	1164	5589	1185	4435	1208	4481	1187	4252
24	14177	1183	6058	1333	5038	1332	5083	1266	4745
25	6889	804	2643	745	2360	752	2367	750	2239
26	9078	965	3550	916	3115	940	3073	922	3003
27	5105	709	1532	606	1446	607	1427	608	1350
28	2574	578	762	432	763	440	777	439	716
29	4083	648	1266	537	1056	536	1089	548	945
30	3087	614	864	475	871	484	875	477	830
31	7150	794	2919	714	2564	719	2573	740	2384
$\Sigma$	388042	34712	152676	38062	130522	38415	131796	38118	122659
$\Sigma(b,w)$	388042	187388		168584		170211		160777	
[%]	207.1	100		89.97		90.83		85.80	

Table 3. Coding results for 'Foreman' at 192 kbps, mean PSNR=30.17.

### Acknowledgements

The authors wish to acknowledge the support of Sony Broadcast and Professional Europe, the CVCP and the University of Bristol for their support of this work.

### References

- [1] Li Wei, Kunt M. *Morphological Segmentation Applied to Displaced Frame Difference Coding*, Signal Processing 38 (1994) pp. 45-56.
- [2] Özcelik T., Katsaggelos A.K. *Detection and Encoding of Model Failures in Very Low Bit Rate Video Coding*, Proc. VCIP'96, SPIE vol. 2727, pp.820-831.
- [3] Witten I.A., Neal R.M., Cleary J.G. *Arithmetic Coding for Data Compression*, Communications of the ACM, vol. 30 no.6 June '87, pp.520-540.
- [4] Le Gall D., Tabatai A. *Sub-band Coding of Images Using Symmetric Short Kernel Filters and Arithmetic Coding Techniques*, Proc. IEEE ICASSP'88, pp. 761-764.
- [5] Czerepinski P.J., Bull D.R., *Morphological Segmentation Based Video Coding Employing Conditional Smoothing*, Proc. IEEE ISCAS'96, pp. 628-631.
- [6] Czerepinski P.J., Bull D.R., *Morphological Video Coder with Conditional Smoothing*, Electronics Letters vol. 32 no.7 (March '96) pp.645-647.